
Introducción a Ruby on Rails

— CSS - Parámetros - Root Path —

Overview

- Agregando estilos básicos a la vista
- Vamos a hacer nuestra aplicación mas dinámica utilizando un parametro de request
- Ruteando un nuevo root path que no sea el que está por defecto.

Layout

app/views/layout/application.html.erb sirve como un contenedor de vistas

```
<!DOCTYPE html>
<html>
<head>
  <title>MyFirstApp</title>
  <%= stylesheet_link_tag "application", media: "all", "data-turbolinks-track" => true %>
  <%= javascript_include_tag "application", "data-turbolinks-track" => true %>
  <%= csrf_meta_tags %>
</head>
<body>
  <%= yield %>
</body>
</html>
```

Cómo luce nuestra aplicación ahora?

Detectando idiomas que pueden interactuar con el Francés - fr

Language	Name
Afrikaans	af
Albanais	sq
Allemand	de
Amharique	am
Anglais	en
Arabe	ar
Arménien	hy
Azéri	az
Basque	eu
Bengali	bn

CSS

▼ my_first_app

- ▼ app
 - ▼ assets
 - ▶ images
 - ▶ javascripts
 - ▼ stylesheets
 - application.css
 - greeter.css.scss
 - translators.css.scss
 - ▶ controllers
 - ▶ helpers
 - ▶ mailers
 - ▶ models
 - ▶ views

```
// Place all the styles related to the translators controller here.  
// They will automatically be included in application.css.  
// You can use Sass (SCSS) here: http://sass-lang.com/
```

```
table{  
  border-collapse: collapse;  
}
```

```
td {  
  padding: 12px;  
}
```

```
.even{  
  background-color: #D6E1C3;  
}
```

cycle helper

```
<h1>Detectando idiomas que pueden interactuar con el Francés - <%= @target_language %></h1>
```

```
<table border="1">
  <tr>
    <th>Language</th>
    <th>Name</th>
  </tr>
  <%= @languages['data']['languages'].each do |lang| %>
    <tr class=<%= cycle('even', 'odd') %>>
      <td><%= lang['name'] %></td>
      <td><%= lang['language'] %></td>
    </tr>
  <%= end %>
</table>
```

Detectando idiomas que pueden interactuar con el Francés - fr

Language	Name
Afrikaans	af
Albanais	sq
Allemand	de
Amharique	am
Anglais	en
Arabe	ar
Arménien	hy
Azéri	az

params helper

- Sería ideal especificar los parámetros que van a los web services.
- Se puede utilizar el hash params para obtener un valor (un parámetro nombrado se vuelve un key o un symbol en el hash)
- Si el parametro no existe, se devuelve nil (comportamiento por defecto del hash)

params helper

```
class TranslatorsController < ApplicationController
  def translate
    @google = GoogleTranslate.new("en", "es")
    @trans = params[:text] || "This is a really good party!"
    @response = @google.translate(@trans)
    @translations = @response['data']['translations'][0]['translatedText']
  end
end
```

< > ↺ localhost:3000/translators/translate?text=I%20love%20my%20course

Translated text!

Me encanta mi curso

Root path

- Qué pasaría si quisiéramos indicar que la primera página será el action de languages? Solo se modifica **routes.rb**

< > ↺ 📄 localhost:3000/?target_language=ar

Idiomas en- ar

Language	Name
الآيسلندية	is
الأذرية	az
الأردية	ur

```
MyFirstApp::Application.routes.draw do
  get "translators/translate"
  get "translators/detect"
  get "translators/languages"

  get 'greeter/hello' => "greeter#hello"
  get 'greeter/goodbye'

  root "translators#languages"
```

Entonces...

- Un pequeño cambio en el **CCS** puede cambiar dramáticamente el aspecto de nuestra aplicación.
- `params` es un helper que permite pasar parámetros en el request (obs: se utiliza en el controller)
- Es fácil cambiar el root path, cambiando en **`routes.rb`**